

基于卡口上下文和深度置信网络的车辆轨迹预测模型研究

李 瞰^{*①} 朱耀堃^① 吴欣虹^① 肖云鹏^① 吴海峰^②

^①(重庆邮电大学软件工程学院 重庆 400065)

^②(海口市气象局 海口 571199)

摘要: 针对车辆轨迹预测中节点序列的时序特性和实际路网中的空间关联性, 该文提出一种基于深度置信网络和SoftMax (DBN-SoftMax) 轨迹预测方法。首先, 考虑到轨迹在节点集合中的强稀疏性和一般特征学习方法对新特征的泛化能力不足, 该文利用深度置信网络(DBN)较强的无监督特征学习能力, 达到提取轨迹局部空间特性的目的; 然后, 针对轨迹的时序特性, 该文采用逻辑回归的预测思路, 用当前轨迹集在路网特征空间中的线性组合来预测轨迹; 最后, 结合自然语言处理领域中的词嵌入的思想, 基于实际轨迹中节点存在的上下文关系, 运用节点的向量集表征了节点间的交通时空关系。实验结果表明该模型不仅能够有效地提取轨迹特征, 并且在拓扑结构复杂的路网中也能得到较好的预测结果。

关键词: 智能交通; 轨迹预测; 卡口上下文分析; 特征提取; 深度置信网络

中图分类号: TP391

文献标识码: A

文章编号: 1009-5896(2021)05-1323-08

DOI: 10.11999/JEIT200137

Vehicle Trajectory Prediction Method Based on Intersection Context and Deep Belief Network

LI Tun^① ZHU Yaokun^① WU Xinhong^① XIAO Yunpeng^① WU Haifeng^②

^①(School of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing 400065, China)

^②(Haikou Meteorological Service, Haikou 571199, China)

Abstract: For the temporal features of trajectory intersection sequence and spatial correlation of the actual road network, a trajectory prediction method based on the Deep Belief Networks and SoftMax (DBN-SoftMax) is proposed. At first, considering the sparsity of trajectory in an intersection set and the insufficiency of generalization ability in general feature learning methods for new features, the strong unsupervised feature learning ability of Deep Belief Network (DBN) is used to extract the local spatial features of trajectory. Secondly, considering the temporal features of the trajectory, the logistic regression method and the linear combination of the current trajectory set in the road network features space are used to predict the trajectory. Finally, Based on the idea of word embedding in the field of natural language processing and the contextual relationship of intersections in the actual trajectory, the vector set of intersections is used to represent the spatiotemporal relationship of traffic between intersections. The experimental results show that the model can not only extract the trajectory features effectively, but also obtain better prediction performance in a road network with complex topology.

Key words: Intelligent traffic; Trajectory prediction; Trajectory intersection context analysis; Feature extract; Deep Belief Network(DBN)

收稿日期: 2020-02-28; 改回日期: 2020-10-15; 网络出版: 2020-10-21

*通信作者: 李瞰 litun@cqupt.edu.cn

基金项目: 国家自然科学基金(61772098), 重庆市教委科技研究项目(KJQN201800641), 重庆邮电大学博士高端人才项目(BYJS2017004), 重庆市技术创新与应用发展专项面上项目(cstc2020jscx-msxmX0150)

Foundation Items: The National Natural Science Foundation of China (61772098), The Science and Technology Research Program of Chongqing Municipal Education Commission (KJQN201800641), The Doctoral Top Talents Program of Chongqing University of Posts and Telecommunications (BYJS2017004), Chongqing Technology Innovation and Application Development Special General Project (cstc2020jscx-msxmX0150)

1 引言

随着城市的迅速发展,交通拥堵逐渐成为人们出行面临的重要问题。如何通过应用技术手段解决这个问题已成为研究热点。车辆轨迹预测不仅可从道路网络的拓扑结构进行分析,还可从车辆轨迹数据中挖掘重要节点^[1]。轨迹预测可以被理解为基于可观察的用户数据和交通网络拓扑来推断可能的缺失轨迹节点与未来轨迹节点。缺失的轨迹节点包含重要的路径信息,因此推断出缺失的轨迹节点具有重要的研究价值。

交通路网中的位置点构成了复杂的网络结构,考虑到道路网络空间中移动物体的属性,文献^[2]定义了基于时间相似性和空间相似性的轨迹表示方式。文献^[3]基于上述表示方法提出了一种基于轨迹数据挖掘方法来预测移动物体位置的模型,具有很好的预测效果。文献^[4]提出了通过隐马尔可夫模型(Hidden Markov Model, HMM)参数空间对单个车辆的轨迹模式进行建模的方法。但由于交通路网中网络复杂的结构造成的用户轨迹模式复杂多样,对大量用户轨迹行为进行建模或者云计算分析需要非常多的资源^[5],通过轨迹模式挖掘用户未来轨迹的方法需要改进。

基于轨迹节点相关性分析的预测是通过分析用户轨迹节点序列之间的相关性来实现的。文献^[6]将用户轨迹与网络轨迹节点时间相似度指标相结合以提高轨迹预测的有效性,尽管该方法很好地利用了轨迹节点的相关性,但是其并没有将时间相关性和空间相关性结合起来,进而忽略了时空特性对用户在选择道路时产生的影响。文献^[7]基于用户群体轨迹在时间和空间上的相似性,以预测节点未来的位置。文献^[8]提出了一种基于特征提取和噪声消除的

交通模式预测框架,通过结合特征提取和噪声消除来提高预测算法的性能。尽管这些方法能够提高数据对轨迹信息的表达能力,但是在预测用户轨迹和分析用户行为模式方面非常耗时。轨迹模式分析中采用聚类的分析思路是当前的一大研究热点^[9,10],文献^[11]提出了一种基于决策图和数据场的轨迹聚类方法,通过自动确定聚类参数并有效地执行轨迹聚类来发现轨迹数据中的热点。文献^[12]提出了一种挖掘双向道路网中密集交通流模式的新技术,通过基于轨迹点密度的聚类发现高密度流量区。

针对上述问题,本文提出了一种基于轨迹特征分析的轨迹预测方法。首先,引入自然语言处理领域中词嵌入的思想提取交通路网中卡口之间的关联性;其次,采用深度置信网络(Deep Belief Network, DBN)学习轨迹矩阵的隐藏特征,使用线性回归网络学习当前已有卡口向量特征与未来卡口向量特征之间的关联关系;最后,在回归预测部分采用了一种权值聚类的方法,使得模型预测的泛化能力有显著提升。

2 模型

本文针对车辆轨迹数据中节点序列的时序特性和交通路网数据中车辆的空间关联性,提出的系统框图如图1所示。首先,根据车辆轨迹数据和交通路网数据进行卡口向量化表示,得到能够表征时空关联性的卡口向量集和用户轨迹向量集;然后,通过轨迹上下文分析将用户轨迹向量集映射到上下文空间,得到影响其未来轨迹选择的影响因子;最后将回归预测与权重聚类相结合,对轨迹进行预测,输出车辆未来位置。

2.1 卡口上下文分析

轨迹数据中最主要的属性就是卡口之间的时空

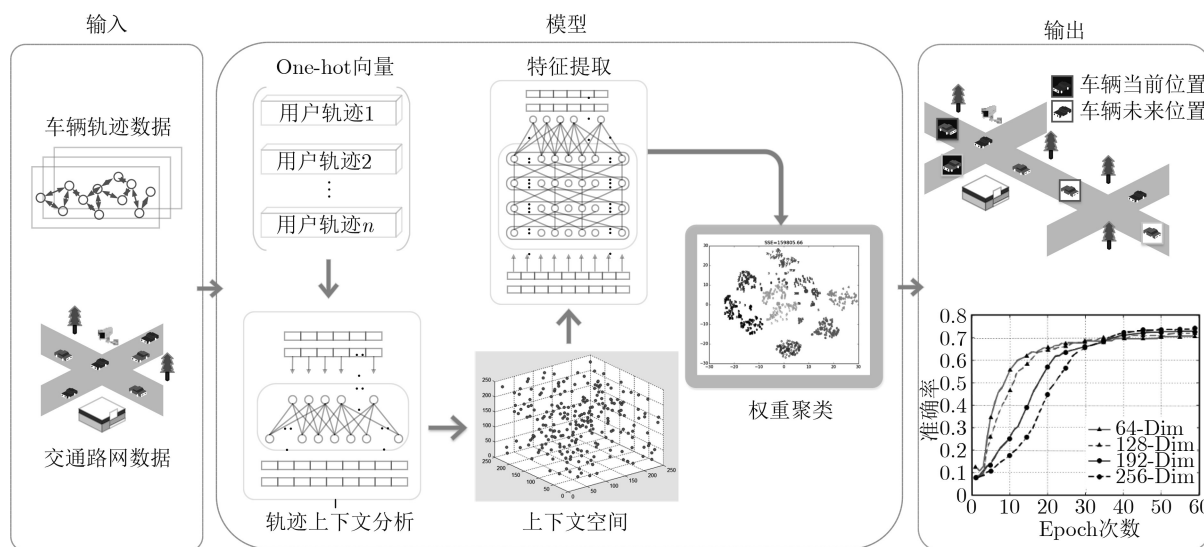


图1 系统框图

关联性，考虑到车辆轨迹中交通卡口之间存在的卡口上下文关系，将用户轨迹作为输入，通过滑动窗口提取卡口的轨迹上下文，再使用One-hot对每一个卡口进行编码，最后使用上下文分析模型得到卡口的上下文语义向量。

2.1.1 统计概率模型

交通卡口与轨迹之间的关系也与自然语言中单词和句子的关系类似，通过统计学概率模型来度量轨迹中卡口之间的关联性，实现使用嵌入化向量表征交通路网中卡口间复杂的时空关联性，定义嵌入化向量的长度为embedSize。使用统计概率模型来计算句子中某一个词语与其他词语的关联性，将其定义为

$$p(\text{Context}(w)|w) = \prod_{u \in \text{Context}(w)} p(u|w) \quad (1)$$

其中， $\text{Context}(w)$ 表示词 w 的上下文，即 w 的周边的词集合，此时， $p(\text{Context}(w)|w)$ 值的大小就能表征词 w 与 $\text{Context}(w)$ 之间关联性的强弱，其表征的是上下文 $\text{Context}(w)$ 出现后，词 w 出现的概率值。与此同时，为了减少计算量，根据n-gram model^[13]中 $n-1$ 阶的Markov假设，认为一个词出现的概率就只与它前面 n 个词相关，即： $\text{Context}(w_i) = \{w_{i-n}, w_{i-n+1}, \dots, w_{i-1}\}$ 。

引入上述模型，交通卡口与其轨迹中其他卡口的关联性可以定义为

$$p(\text{Context}(c)|c) = \prod_{u \in \text{Context}(c)} p(u|c) \quad (2)$$

其中， $\text{Context}(c)$ 表示轨迹集 T 中，与卡口 c 出现在同一条轨迹 t_j 的前后共 $2n$ 个卡口，即 $\text{Context}(c_i) = \{c_{i-n}, c_{i-n+1}, \dots, c_{i-1}, c_{i+1}, \dots, c_{i+n-1}, c_{i+n}\}, c_j, c \in t_j$ 。

接着，利用最大对数似然对轨迹集建模，设计了一个目标函数

$$L = \lg p(\text{Context}(c)|c) \quad (3)$$

在此模型中，对于每一条轨迹里面的所有卡口 c ，均都希望 $p(\text{Context}(c)|c)$ 达到最大，进而对于整个数据集来说，最大化目标函数 L 便成了统计概率模型对轨迹数据集进行建模的首要目标。

2.1.2 卡口上下文分析模型

模型使用基于Negative Sampling的Skip-gram策略^[14]对上述网络进行计算，已知卡口 c_0 ，需要预测 $\text{Context}(c_0)$ ，因此对于给定的卡口上下文 $\text{Context}(c_0)$ ，卡口 c_0 就是一个正样本，其他不在 $\text{Context}(c_0)$ 中的卡口就是负样本，通过采样的方法可采样出neg个负样本，得到一个训练样本 $(\text{Context}(c_0), c_i)$ ， $i = 0, 1, 2, \dots, \text{neg}$ ；其中， $i = 0$ 设为正例， $i = 1, 2, \dots, \text{neg}$ 设为负例。如图1所示，该网络的输入为所要求

的嵌入化向量，对于训练样本 $(\text{Context}(c_0), c_i)$ ， $i = 0, 1, 2, \dots, \text{neg}$ ，定义 x_{c_0} 为嵌入化向量作为网络的输入， y_0 作为网络的输出， $y_0 = 1$ 表示正例， $y_0 = 0$ 表示负例。于是，可将前文所提到的卡口上下文统计概率值的计算定义为关于 $\text{Context}(c_0)$ 和 c_0 的函数，即

$$F(c_0, \text{Context}(c_0), \theta) = p(\text{Context}(c_0)|c_0) \quad (4)$$

其中， $\theta = [\theta_{c_1}, \theta_{c_2}, \dots, \theta_{c_m}]$ ， $\theta_{c_i} \in R^{\text{embedSize}}$ ，采用2元逻辑回归来求解模型参数 θ ，即正例应该满足

$$p(c_i, \text{Context}(c_0)) = \sigma(\mathbf{x}_{c_0}^T \theta_{c_i}), y_i = 1, i = 0 \quad (5)$$

负例应该满足

$$p(c_i, \text{Context}(c_0)) = 1 - \sigma(\mathbf{x}_{c_0}^T \theta_{c_i}), y_i = 0, \\ i = 1, 2, \dots, \text{neg} \quad (6)$$

$F(c_0, \text{Context}(c_0), \theta)$ 应该满足

$$F(c_0, \text{Context}(c_0), \theta) = \prod_{i=0}^{\text{neg}} \sigma(\mathbf{x}_{c_0}^T \theta_{c_i})^{y_i} (1 - \sigma(\mathbf{x}_{c_0}^T \theta_{c_i}))^{1-y_i} \quad (7)$$

此时对应的目标函数为

$$L = \sum_{i=0}^{\text{neg}} y_i \lg \sigma(\mathbf{x}_{c_0}^T \theta_{c_i}) + (1 - y_i) \lg (1 - \sigma(\mathbf{x}_{c_0}^T \theta_{c_i})) \quad (8)$$

采用随机梯度上升法，对 $\mathbf{x}_{c_0}, \theta_{c_i}, i = 0, 1, 2, \dots, \text{neg}$ 进行迭代求解，分别计算它们的梯度

$$\frac{\partial L}{\partial \theta_{c_i}} = y_i (1 - \sigma(\mathbf{x}_{c_0}^T \theta_{c_i})) \mathbf{x}_{c_0} + (1 - y_i) \sigma(\mathbf{x}_{c_0}^T \theta_{c_i}) \mathbf{x}_{c_0} \quad (9)$$

$$\frac{\partial L}{\partial \theta_{c_i}} = (y_i - \sigma(\mathbf{x}_{c_0}^T \theta_{c_i})) \mathbf{x}_{c_0} \quad (10)$$

$$\frac{\partial L}{\partial \mathbf{x}_{c_0}} = \sum_{i=0}^{\text{neg}} (y_i - \sigma(\mathbf{x}_{c_0}^T \theta_{c_i})) \theta_{c_i} \quad (11)$$

因此，对于所有卡口 $c_i \in C$ ，可求得所需的嵌入化向量 \mathbf{x}_{c_i} ，并且对于每一条轨迹 $t_i = [c_1, c_2, \dots, c_n]^T$ ，即可以得到其嵌入化向量表示 $\mathbf{t}'_i = [\mathbf{x}_{c_1}, \mathbf{x}_{c_2}, \dots, \mathbf{x}_{c_n}]^T$ 。

2.2 预测模型

近年来，许多深度学习的研究表明，在对图像和音频的分类、识别等任务中，可通过分层训练神经网络来获得更好的效果^[15-17]。DBN是深度学习模型中常用和有效的方法之一，它是一堆受限玻尔兹曼机(Restricted Boltzmann Machine, RBM)，每个RBM只有一个隐藏层，学习单元对一个RBM的激活被用作堆栈中下一个RBM的输入数据。Hinton等人^[15]提出了一种快速贪婪学习DBN的方法，该方法1次学习1层。

RBM是一种特殊类型的马尔科夫随机场，它是一个无向图模型，其中可见变量 v 通过无向加权

与随机隐藏单元 h 连接。由于隐藏变量或可见变量之间没有连接,所以它们是受限的。该模型通过一个能量函数 $E = (v, h; \theta)$ 定义了 v, h 上的概率分布。假设它是一个2元RBM,它可以写成

$$-\lg P(\mathbf{v}, \mathbf{h}) \propto E(\mathbf{v}, \mathbf{h}; \theta) = -\sum_{i=1}^{|\mathbf{V}|} \sum_{j=1}^{|\mathbf{H}|} \mathbf{w}_{ij} \mathbf{v}_i \mathbf{h}_j - \sum_{i=1}^{|\mathbf{V}|} b_i \mathbf{v}_i - \sum_{j=1}^{|\mathbf{H}|} a_j \mathbf{h}_j \quad (12)$$

其中, $\theta = (w, b, a)$ 是参数集, $w_{i,j}$ 是连接可见层单元 i 和隐藏层 j 的权重矩阵,而 b_i 和 a_j 是其偏置,可见层与隐藏层的神经元数量用 $|\mathbf{V}|$ 和 $|\mathbf{H}|$ 来表示。当 v 或 h 固定时,条件概率可以计算为

$$p(\mathbf{h}_j | \mathbf{v}; \theta) = \text{sigm} \left(\sum_{i=1}^{|\mathbf{V}|} \mathbf{w}_{ij} \mathbf{v}_i + a_j \right) \quad (13)$$

$$p(\mathbf{v}_i | \mathbf{h}; \theta) = \text{sigm} \left(\sum_{j=1}^{|\mathbf{H}|} \mathbf{w}_{ij} \mathbf{h}_j + b_i \right) \quad (14)$$

其中, $\text{sigm}(x) = 1/(1 + e^{-x})$ 是Sigmoid函数,参数集 $\theta = (w, b, a)$ 中的参数可以使用对比散度进行有效的学习。

然后,将几个RBM一起堆叠成一个DBN来进行训练,其关键思想是,由于RBM学习的参数 θ 是由 $p(\mathbf{v} | \mathbf{h}, \theta)$ 和先验分布 $p(\mathbf{h} | \theta)$ 共同决定的。因此,生成可见层的概率可写为

$$p(\mathbf{v}) = \sum_{\mathbf{h}} p(\mathbf{h} | \theta) p(\mathbf{v} | \mathbf{h}, \theta) \quad (15)$$

RBM学习 θ 之后,得到了 $p(\mathbf{v} | \mathbf{h}, \theta)$,并且 $p(\mathbf{h} | \theta)$ 可以被一个连续的RBM替代,该RBM将前一个RBM的隐藏层视为可见层,这样,DBN就可以将下一个RBM层的隐藏层作为新的特征层进行学习,此方法可作为一个无监督的特征学习方法。

图2展示了本文对车辆轨迹进行预测的深层网络架构。输入空间 X 是一条轨迹中所有轨迹节点的

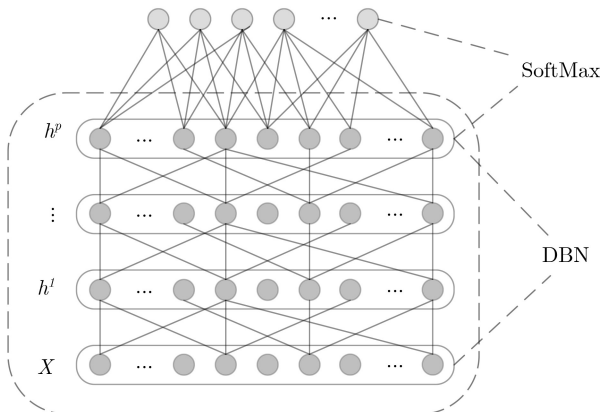


图2 DBN-SoftMax网络结构示意图

嵌入化向量 $\mathbf{t}'_r = [\mathbf{x}_{c1}, \mathbf{x}_{c2}, \dots, \mathbf{x}_{cn}]^T$ 首尾拼接得到的向量 $\mathbf{v}_r = [\mathbf{v}_{r,1}, \mathbf{v}_{r,2}, \dots, \mathbf{v}_{r,n \times \text{embedSize}}]$,通过输入到DBN网络,预测该轨迹下一个时刻的轨迹节点 \mathbf{c}_{n+1} 。首先,使用 \mathbf{v}_r 作为第1层RBM的可见层,进行计算为

$$p(\mathbf{h}_{r,j} | \mathbf{v}_r; \theta) = \text{sigm} \left(\sum_{i=1}^{|\mathbf{v}_r|} \mathbf{w}_{ij} \mathbf{v}_{r,i} + a_j \right) \quad (16)$$

得到隐藏层的向量 \mathbf{h}_r ,进行计算更新 \mathbf{v}_r

$$p(\mathbf{v}_{r,i} | \mathbf{h}_r; \theta) = \text{sigm} \left(\sum_{j=1}^{|\mathbf{h}_r|} \mathbf{w}_{ij} \mathbf{h}_{r,j} + b_i \right) \quad (17)$$

反复进行上述两步计算,经过多层的RBM可见层与隐藏层的学习,利用对比散度学习参数集 $\theta = \{w, a, b\}$,并采样出最后一层RBM的隐藏层作为顶层的特征层 \mathbf{h}^p ,接着在顶层使用SoftMaxLoss和反向传播算法对DBN网络参数进行有监督的调整,具体为

$$y_r = g(\mathbf{w}_r \cdot \mathbf{h}^p + b_i) \quad (18)$$

$$g(x_i) = \frac{e^{x_i}}{\sum_{c=1}^C e^{x_c}} \quad (19)$$

其中, $g(\cdot)$ 是SoftMax函数, $y_r \in R^m$ 为轨迹 \mathbf{v}_r 下一个时刻的轨迹节点在整个节点集 C 的概率分布,利用反向传播算法更新权重矩阵 \mathbf{w}_r ,直到得到权重矩阵集合 $\mathbf{w} = \{\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_m\}$,其中, $\mathbf{w}_i \in R^{m \times k}$ 为预测模型对第 i 条轨迹的权重矩阵,对 \mathbf{w} 进行无监督训练,得到聚类后的权重集合 $\mathbf{w}' = \{\mathbf{w}'_1, \mathbf{w}'_2, \dots, \mathbf{w}'_s\}$ 。

在模型的真实预测阶段,使用SoftMax分类器来做最终的预测,具体为

$$\text{out} = \arg \max_{\text{out}} (\mathbf{O}) \quad (20)$$

其中, $\mathbf{O} = g(\mathbf{w}'_i \cdot \mathbf{h}^p_i) = [\mathbf{o}_1, \mathbf{o}_2, \dots, \mathbf{o}_{\text{out}}, \dots, \mathbf{o}_m]$, $\mathbf{o}_{\text{out}} \in (0, 1)$, $\text{out} = \{1, 2, \dots, m\}$, out 即为所预测的卡口轨迹点的编号。

2.3 学习算法

实验过程主要分为两步:首先,对输入的所有交通卡口进行卡口嵌入化,获取该模块的参数结果 θ, x ;其次,将上一步中的结果和所有交通轨迹集作为输入,进行对DBN-SoftMax模型的训练,得到最后的模型参数 $\mathbf{w}' = \{\mathbf{w}'_1, \mathbf{w}'_2, \dots, \mathbf{w}'_s\}$,用于对轨迹的下一个节点进行预测,具体的学习算法如表1所示。

本算法中,在卡口嵌入化阶段(intersections embedding)的时间复杂度为 $O(N)$;在路网特征提

表 1 DBN-SoftMax算法

输入:
交通卡口集 $C = \{c_1, c_2, \dots, c_m\}$;
交通轨迹 $T = \{t_1, t_2, \dots, t_m\}$, $t_i = [c_1, c_2, \dots, c_m]^T$;
输出:
预测卡口编号;
初始化 $\theta = [\theta_{c_1}, \theta_{c_2}, \dots, \theta_{c_m}]$, $x = [x_{c_1}, x_{c_2}, \dots, x_{c_m}]$
采样 neg 个负样本 $(Context(c_0), c_0)$ 对每一个采样
$((Context(c_0), c_i), i = 0, 1, 2, \dots, neg)$
对每一个采样 $((Context(c_0), c_i), i = 0, 1, 2, \dots, neg)$ do :
for $i = 1$ to $2c$:
det $aX = 0$
for $j = 0$ to net :
计算 $f = \sigma(x_{c_0}^T \theta_{c_j})$
计算 $g = (y_i - f)\eta$
更新 $detaX = detaX + g\theta_{c_j}$
更新 $\theta_{c_j} = \theta_{c_j} + gx_{c_0}$
end for
更新 $x_{c_0} = x_{c_0} + detaX$
end for
获取参数 θ, x
嵌入化轨迹集 $t' = \{t'_1, t'_2, \dots, t'_m\}$, $t'_i = [x_{c_1}, x_{c_2}, \dots, x_{c_m}]^T$
嵌入化向量 t'_i 首尾拼接得到 v_i
使用 CD-k 算法实现 RBM
DBN 深度为 DEPTH, RBM 隐藏层的大小为 RBM-SIZE
初始化 $\theta = \{\theta_1, \theta_2, \dots, \theta_{DEPTH}\}$
初始迭代
for i in DEPTH, do
for 在范围 ITERATION 内迭代, do :
使用 CD-k 算法:
通过式(16)计算概率分布 $p(h_i v_i; \theta_i)$
通过式(17)计算概率分布 $p(v_i h_i; \theta_i)$
采样获取 h_i 和 θ_i
产生 $\Delta w_i, \Delta b_i$
更新 $w_i = w_i + \Delta w_i, b_i = b_i + \Delta b_i$
End for
End for
权重矩阵 $w = \{w_1, w_2, \dots, w_m\}$ 聚类得到 $w' = \{w'_1, w'_2, \dots, w'_m\}$
使用式(20)预测下一个卡口编号

取阶段(DBN-SoftMax model)的时间复杂度为 $O(N \lg(N))$; 在权重聚类阶段(weight clustering), K-means的时间复杂度为 $O((k \times d \times i)N)$, 其中, k 为聚类中心数量, d 为权重 w_i 的维度, i 为迭代次数。因此, 通过以上分析, 整个算法的时间复杂度为 $O(N) + O(N \lg(N)) + O((k \times d \times i)N) \sim O(N \lg(N))$ 。

3 实验结果及分析

3.1 数据集

本文实验所使用的数据为中国某省会城市监测设备采集的真实数据, 包括从2017年9月—2017年11月该城市路网交通卡口所监控的真实过车记录。在数据量方面, 该城市每天产生的过车数据基本在 10^6 级别, 经过初级的处理之后得到结构化的数据, 包括车牌号、过车时间、交通卡口编号、交通卡口的经度和纬度等信息。与传统的GPS车辆位置数据不同, 此数据集为定点拍摄, 车辆在固定位置被记录, 不用再进行热点位置聚类。数据样例如表2表示, 由于数据敏感性, 经纬度数据不做展示。

为了使轨迹数据能够真实反映交通路网的时空关系以及车辆用户行车习惯, 本实验针对数据特性做了预处理。首先, 由于拍摄数据存在一些重复、冗余的轨迹数据, 对这部分数据进行了去重处理; 其次, 针对少数不良行车行为, 如套牌车现象, 对每条轨迹内的卡口位置进行速度筛选。通过将不良行车数据进行过滤, 获得数据集包括176000条轨迹, 轨迹包含6到10个交通卡口不等, 总共965个交通卡口, 采用留出法将数据集划分互不相交的训练集和测试集, 训练集为140800条轨迹, 测试集为剩下的35200条轨迹。

3.2 算法预测性能

3.2.1 不同嵌入化维度下的性能对比

为了比较不同的交通卡口嵌入化维度在预测模型(DBN-SoftMax)中的表现, 分别对比了不同的上下文长度下不同的嵌入化维度的性能, 以选取最佳的嵌入化维度和上下文长度, 并从侧面验证了本文提出的交通卡口表示方法对轨迹预测的有效性, 结果如图3所示。

图3展示了嵌入化维度和上下文长度两个参数的选取对模型最后准确率的影响, 由图3中可以看出, 在对路网卡口进行建模时, 选取不同的嵌入化维度对模型训练的收敛速度有很大的影响, 同时, 也对最后的准确率有一定的影响。嵌入化维度越大, 准确率越高, 使用卡口上下文向量来提取轨迹中节点之间的语义关系具有一定的作用, 且语义向量的长度能够体现语义特征的容量, 预测性能较好。

从图4可知, 当嵌入化维度取值为128维时, 不同的上下文长度的准确率(accuracy)和训练时间(training time), 随着上下文长度的增加都在增加, 但是随着上下文长度的增加, 模型训练时间增幅比较大, 所以选择上下文长度为4可以得到较好的实验表现。

表2 数据样例

车牌号	通过时间	卡口号	纬度	经度
*A***14B	2017/09/09 09:09	1534****798	Lat1	Lng1
*D***153	2017/09/14 14:49	JF4****798	Lat2	Lng2
*B***433	2017/10/02 09:35	1534****342	Lat3	Lng3
*C***545	2017/10/24 09:51	JF4****798	Lat4	Lng4
*A***900	2017/11/09 13:54	SH****798	Lat5	Lng5
*E***M78	2017/11/27 18:06	1534****092	Lat6	Lng6

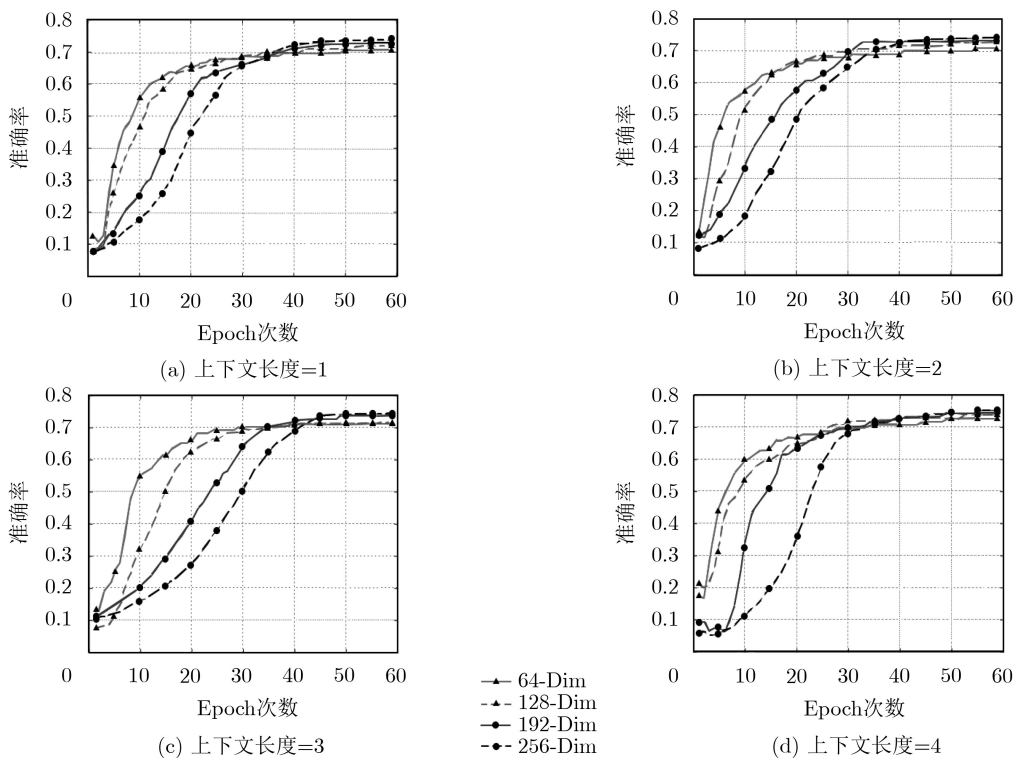


图3 对比不同上下文长度下、不同嵌入化维度下的模型准确率

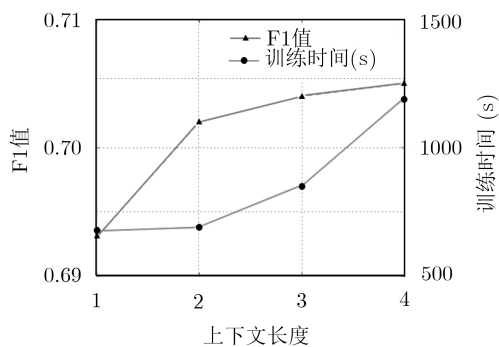


图4 不同上下文长度下模型的F1值和训练时间

3.2.2 不同预测策略下的性能表现对比

对上一节中轨迹节点嵌入化维度数据, 分别使用DBN-SoftMax, NN-SoftMax和RBF SVM模型进行性能分析对比。并对比了不同的DBN层数在轨迹预测任务中的表现, 以选取最佳的DBN深度(depth)参数。选取嵌入化维度为128, 训练集的轨迹长度为6的轨迹数据进行对比试验, 结果如表3所示。

表3展示了模型预测结果的准确率(accuracy)、召回率(recall)、F1值、训练时间和测试时间, 其中表现性能最好的模型已通过加黑体标识, 即DBN-SoftMax层数为4层时。上述实验表明, DBN网络层数的加深能够对轨迹预测任务的性能有显著的提升; 并且, 使用1层的DBN作为特征提取网络, 能够显著超过4层前馈神经网络所达到的预测效果。另外, 从表3中可以看出, DBN-SoftMax和NN-SoftMax相对于RBF SVM在模型的训练上耗费了太多时间, 但是, 其在新数据进行预测分析时的速度非常快, 这点在模型应用方面有巨大优势。

3.2.3 不同算法的评估对比

基于以上的参数选取以及实验数据集, 使用Markov^[18], LSTM^[19]和MMM^[20]基线算法与本文所提算法进行对比, 利用评价指标对上述4个算法进行了评估, 选取近176000条长度为4的交通车辆真实轨迹进行下一个位置点的轨迹预测, 各算法实验效果如图5所示。

表 3 实验结果指标对比

分类器	深度	精准率	召回率	F1值	训练时间(s)	测试时间(s)
DBN-SoftMax	1	0.693	0.672	0.6823	675.178	0.001
	2	0.704	0.686	0.6948	792.213	0.014
	3	0.709	0.693	0.7009	903.291	0.016
	4	0.715	0.702	0.7084	1093.629	0.020
	5	0.710	0.701	0.7055	1309.841	0.023
NN-SoftMax		0.691	0.684	0.6875	939.620	0.031
RBF SVM		0.681	0.665	0.6729	103.897	98.460

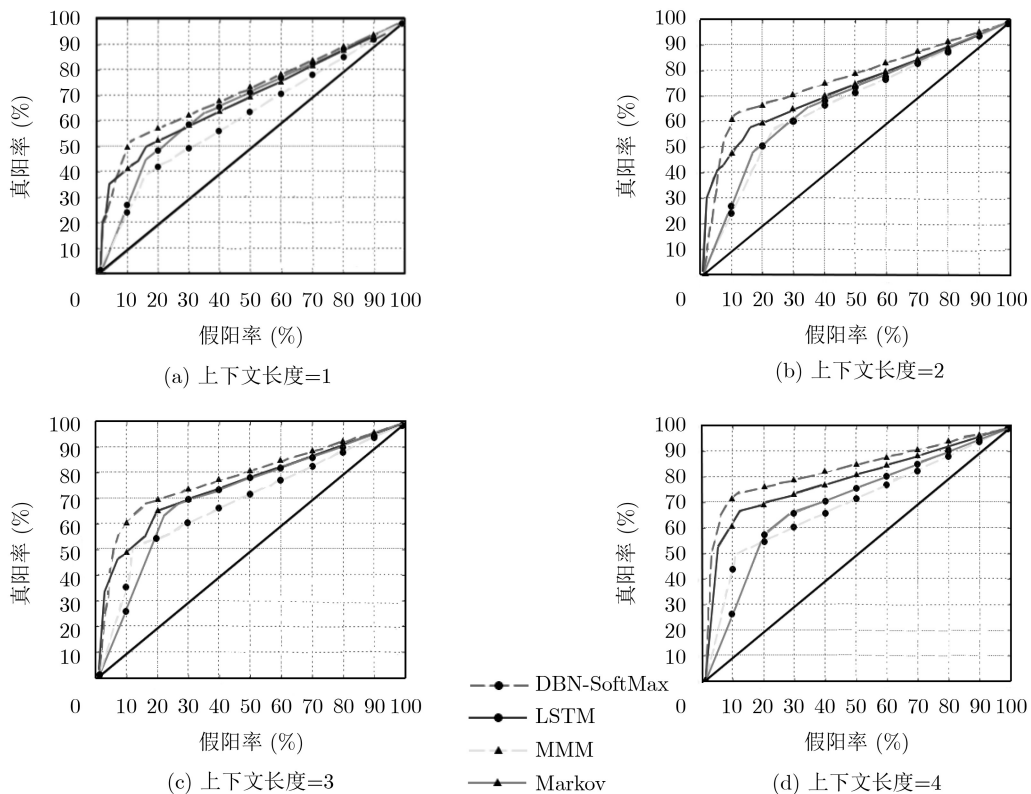


图 5 对比不同上下文长度下不同算法的ROC曲线

从图5可知，在不同的上下文条件下，本文提出的DBN-SoftMax算法的ROC曲线更靠近左上，这意味着本文所提算法对车辆轨迹预测获得了更好的结果。DBN-SoftMax算法因其其在轨迹特征方面的优秀性能而具有较高的预测准确率，并且在不同的历史轨迹下均具有良好表现，而其他基线算法在较少的历史轨迹作为数据时表现欠佳。基于卡口上下文分析的车辆轨迹预测模型相比其他算法具有更好的预测效果，并且在历史信息较少的前提下，算法仍能具有较好的表现。

4 结束语

本文利用交通轨迹数据中轨迹节点存在的上下文关系特性，提出一种基于对卡口上下文进行特征提取的交通车辆轨迹预测的方法。首先，利用实际

轨迹中节点存在的上下文关系，构建轨迹节点的向量空间，运用节点的向量集表征节点间的交通时空关系；其次，该模型利用DBN提取轨迹局部空间特性；最后，该模型使用权重聚类，对结果进行了优化。实验结果表明该模型不仅能够有效地提取轨迹特征，并且在拓扑结构复杂的路网中也能得到较好的预测结果。在未来的工作中，将考虑更多、更复杂的数据对轨迹预测效果的影响，如果能够同时采集到车辆用户数据，将结合用户个人信息与交通路网数据进行分析建模，从而更加准确地预测车辆轨迹。

参考文献

[1] 芮兰兰, 李钦铭. 基于组合模型的短时交通流量预测算法[J]. 电子与信息学报, 2016, 38(5): 1227-1233. doi: 10.11999/JEIT150846.

- RUI Lanlan and LI Qinning. Short-term traffic flow prediction algorithm based on combined model[J]. *Journal of Electronics & Information Technology*, 2016, 38(5): 1227–1233. doi: [10.11999/JEIT150846](https://doi.org/10.11999/JEIT150846).
- [2] YUAN Guan, SUN Penghui, ZHAO Jie, *et al.* A review of moving object trajectory clustering algorithms[J]. *Artificial Intelligence Review*, 2017, 47(1): 123–144. doi: [10.1007/s10462-016-9477-7](https://doi.org/10.1007/s10462-016-9477-7).
- [3] ENDO Y, TODA H, NISHIDA K, *et al.* Deep feature extraction from trajectories for transportation mode estimation[C]. The 20th Pacific-Asia Conference on Knowledge Discovery and Data Mining, Auckland, New Zealand, 2016: 54–66. doi: [10.1007/978-3-319-31750-2_5](https://doi.org/10.1007/978-3-319-31750-2_5).
- [4] PORIKLI F. Clustering variable length sequences by eigenvector decomposition using HMM[C]. The Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR), Lisbon, Portugal, 2004: 352–360. doi: [10.1007/978-3-540-27868-9_37](https://doi.org/10.1007/978-3-540-27868-9_37).
- [5] WU Bin and QIN Lei. Design and implementation of business-driven bi platform based on cloud computing[C]. 2011 IEEE International Conference on Cloud Computing and Intelligence Systems, Beijing, China, 2011: 118–122. doi: [10.1109/CCIS.2011.6045044](https://doi.org/10.1109/CCIS.2011.6045044).
- [6] WANG Yuqi, CAO Jiannong, LI Wengen, *et al.* Exploring traffic congestion correlation from multiple data sources[J]. *Pervasive and Mobile Computing*, 2017, 41: 470–483. doi: [10.1016/j.pmcj.2017.03.015](https://doi.org/10.1016/j.pmcj.2017.03.015).
- [7] ANAGOSTOPOULOS C and HADJIEFTHYMIADES S. Intelligent trajectory classification for improved movement prediction[J]. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2014, 44(10): 1301–1314. doi: [10.1109/TSMC.2014.2316742](https://doi.org/10.1109/TSMC.2014.2316742).
- [8] ZHANG Fusang, JIN Beihong, WANG Zhaoyang, *et al.* On geocasting over urban bus-based networks by mining trajectories[J]. *IEEE Transactions on Intelligent Transportation Systems*, 2016, 17(6): 1734–1747. doi: [10.1109/TITS.2015.2504513](https://doi.org/10.1109/TITS.2015.2504513).
- [9] 陈忠辉, 凌献尧, 冯心欣, 等. 基于模糊C均值聚类和随机森林的短时交通状态预测方法[J]. *电子与信息学报*, 2018, 40(8): 1879–1886. doi: [10.11999/JEIT171090](https://doi.org/10.11999/JEIT171090).
- CHEN Zhonghui, LING Xianyao, FENG Xinxin, *et al.* Short-term traffic state prediction approach based on FCM and random forest[J]. *Journal of Electronics & Information Technology*, 2018, 40(8): 1879–1886. doi: [10.11999/JEIT171090](https://doi.org/10.11999/JEIT171090).
- [10] PIRES T J P and FIGUEIREDO M A T. Shape-based trajectory clustering[C]. The 6th International Conference on Pattern Recognition Applications and Methods, Porto, Portugal, 2017: 71–81. doi: [10.5220/0006117400710081](https://doi.org/10.5220/0006117400710081).
- [11] ZHAO Pengxiang, QIN Kun, YE Xinyue, *et al.* A trajectory clustering approach based on decision graph and data field for detecting hotspots[J]. *International Journal of Geographical Information Science*, 2017, 31(6): 1101–1127. doi: [10.1080/13658816.2016.1213845](https://doi.org/10.1080/13658816.2016.1213845).
- [12] MIRGE V, VERMA K, and GUPTA S. Dense traffic flow patterns mining in bi-directional road networks using density based trajectory clustering[J]. *Advances in Data Analysis and Classification*, 2017, 11(3): 547–561. doi: [10.1007/s11634-016-0256-8](https://doi.org/10.1007/s11634-016-0256-8).
- [13] BROWN P F, DESOUZA P V, MERCER R L, *et al.* Class-based n -gram models of natural language[J]. *Computational Linguistics*, 1992, 18(4): 467–479.
- [14] MIKOLOV T, SUTSKEVER I, CHEN Kai, *et al.* Distributed representations of words and phrases and their compositionality[C]. The 26th International Conference on Neural Information Processing Systems, Lake Tahoe, United States, 2013: 3111–3119.
- [15] HINTON G E, OSINDERO S, and TEH Y W. A fast learning algorithm for deep belief nets[J]. *Neural Computation*, 2006, 18(7): 1527–1554. doi: [10.1162/neco.2006.18.7.1527](https://doi.org/10.1162/neco.2006.18.7.1527).
- [16] KRIZHEVSKY A, SUTSKEVER I, and HINTON G E. ImageNet classification with deep convolutional neural networks[C]. The 25th International Conference on Neural Information Processing Systems, Lake Tahoe, USA, 2012: 1097–1105.
- [17] LAROCHELLE H, BENGIO Y, LOURADOUR J, *et al.* Exploring strategies for training deep neural networks[J]. *The Journal of Machine Learning Research*, 2009, 10(1): 1–40.
- [18] MENZ L, HERBERTH R, LUO Chunbo, *et al.* An improved method for mobility prediction using a Markov model and density estimation[C]. 2018 IEEE Wireless Communications and Networking Conference, Barcelona, Spain, 2018: 1–6. doi: [10.1109/WCNC.2018.8377086](https://doi.org/10.1109/WCNC.2018.8377086).
- [19] XUE Hao, HUYNH D Q, and REYNOLDS M. SS-LSTM: A hierarchical LSTM model for pedestrian trajectory prediction[C]. 2018 IEEE Winter Conference on Applications of Computer Vision, Lake Tahoe, USA, 2018: 1186–1194. doi: [10.1109/WACV.2018.00135](https://doi.org/10.1109/WACV.2018.00135).
- [20] GIACOMETTI A and SOULET A. Frequent pattern outlier detection without exhaustive mining[C]. The 20th Pacific-Asia Conference on Knowledge Discovery and Data Mining, Auckland, New Zealand, 2016: 196–207. doi: [10.1007/978-3-319-31750-2_16](https://doi.org/10.1007/978-3-319-31750-2_16).
- 李 瞰: 男, 1980年生, 高级工程师, 研究方向为机器学习和网络安全。
- 朱耀堃: 男, 1994年生, 硕士, 研究方向为智慧交通、人工智能。
- 吴欣虹: 女, 1995年生, 硕士生, 研究方向为社交网络、机器学习。
- 肖云鹏: 男, 1979年生, 教授, 研究方向为大数据、移动互联网、信息安全。
- 吴海峰: 男, 1978年生, 助理工程师, 研究方向为计算机网络。